

GENERATING DISASSEMBLY SEQUENCES OF MECHANICAL ASSEMBLY FROM STEP-CAD MODEL

GIRI. R

Department of Mechatronics Engineering, Rajalakshmi Engineering College, Chennai, Tamil Nadu, India

ABSTRACT

Lot of research works have been made on identifying feasible disassembly sequences for an assembly. However, those research works were concentrated on an assembly in which the parts must have single step to disassemble. This paper proposes a novel methodology to develop the feasible disassembly sequences without human interaction for a mechanical assembly in which the disassembly operation requires one or two direction movements.

KEYWORDS: CAD, STEP file, Disassembly Sequence Generations & Genetic Algorithm

Received: Feb 25, 2019; **Accepted:** Mar 18, 2019; **Published:** Apr 19, 2019; **Paper Id.:** IJMPERDJUN201940

INTRODUCTION

The disassembly operation is the process of separating a part from the assembly and is carried out for the purpose of accessing components for maintenance, replacement or repair. The disassembly sequence has a direct effect on the efficiency of the disassembly process and the choice of the sequence in which parts or subassemblies are put together in the mechanical assembly of a product can drastically affect the efficiency of the assembly process (Homem de Mello and Sanderson, 1991). Bourjault (1984) proposed a method to generate assembly sequences based on the answers to a set of structured questions. The number of questions to be answered by the programmer in Bourjault's procedure increases exponentially as the part number in the assembly increases. DeFazio and Whitney (1987) simplified Bourjault's procedure by reducing the questions. Homem de Mello and Sanderson (1990) developed AND/OR graph to generate all assembly sequences from the feasible cut sets of the part connection graph. A few researchers have developed connector based algorithm for assembly sequencing generation (Tseng and Li, 1999; Yin et al., 2003; Li et al., 2012). The research works carried out for identifying assembly/disassembly sequences using connectors have not considered all fasteners. Many researchers used CAD files for generating assembly and disassembly sequences (Rabemanantsoa and Pierce, 1996; Mok et al., 2001; Sung, 2001; Gottipolu and Ghosh, 2003; Zha, 2004; Li et al., 2012; Vigano and Osorio-Gomez, 2011). Sung (2001) developed an assembly sequence generation by using the VRML files and the contact between the parts are identified by using the octree representation method. Zha (2004) developed an approach for assembly planning using STEP-based product model to integrate the parts list, assembly structures, mating relationships and features of solid models of component. Some researchers developed graphs for disassembly sequence planning (Ong and Wong, 1999; Kuo, 2000; Li et al., 2002; Xing et al., 2007). But, the graphical approach requires more time to find the solution. Genetic algorithm (Smith et al., 2002; Pan et al., 2006; Smith et al., 2012), Artificial neural network (Cem et al., 2005), Ant colony system (Huan & Xu., 2017), Particle swarm algorithm (Raju. et al., 2016) and Bat algorithm (Murali et al., 2018) were also used for assembly/disassembly sequence generation. The purpose of this study is to

enhance the cutest method of the contact graph used to identify the disassembly sequences since the number of cutest to be tested increases exponentially when the parts in assembly increases. In this method, an attempt is made by using genetic algorithm to identify the removal directions for a part which requires either one or two steps to disassemble.

Nomenclature

TC_1	Edge value
TC_2	Edge cut penalty
TC_3	Precedence violation value
n	Number of edges in the graph
$a1$	Edge value constant
x	The chromosome of size $(1, n)$
$s1$	String position during the edge cut
$a2$	Edge cut constant
N	Population size
a	The row in the interference matrix
d	The column in the interference matrix
$((\exists d)T(f, a)) = 0$	The interference matrix element value of part f in the row a is zero in any one column d
$\deg(p_i)$	Degree of node p_i
(p_i, p_j)	Vertices of any edge in part contact graph

OUTLINE OF THE PROPOSED SYSTEM

The flowchart of the proposed method is shown in figure 1. Initially, the assembly is modelled in CAD software and converted into the STEP file format. STEP file is a neutral file which acts as the focal point of communication among the dissimilar database structures of CAD/CAM systems. STEP is an ISO standard structure and the files are fully interpretable by computer. There are several sections within STEP, called Application Protocols (APs) which are built into a common data model. STEP AP203 format is used to save solid models. The transformation matrix of each part with respect to the universal Coordinate system can be obtained from STEP assembly file. A face in STEP file is represented as ADVANCED_FACE entity. FACE_OUTER_BOUND and FACE_BOUND are sub entities of the ADVANCED_FACE. FACE_OUTER_BOUND is the outer edge loop of a face whereas FACE_BOUND is the inner loop of the face. EDGE_LOOP has the collection of closed edge curves that forms the face. ORIENTED_EDGE is the sub entity of EDGE_LOOP, whereas the EDGE_CURVE is the sub entity of ORIENTED_EDGE. EDGE_CURVE is an entity which connects two vertex points by a line or arc. The cylindrical object shown in figure 2 is represented in STEP format as four edge loops. Edge loop 3 and 4 represent the faces as cylindrical. Edge loop 1 and 2 are represented the top and bottom circle. In the second step of the proposed method, the faces of each part are extracted and their tessellated triangular facets are stored. In the third step, rays are fired from the vertices of each tessellated triangle of a part with other part. The part contact graph is formulated from the results obtained from the ray tracing algorithm.

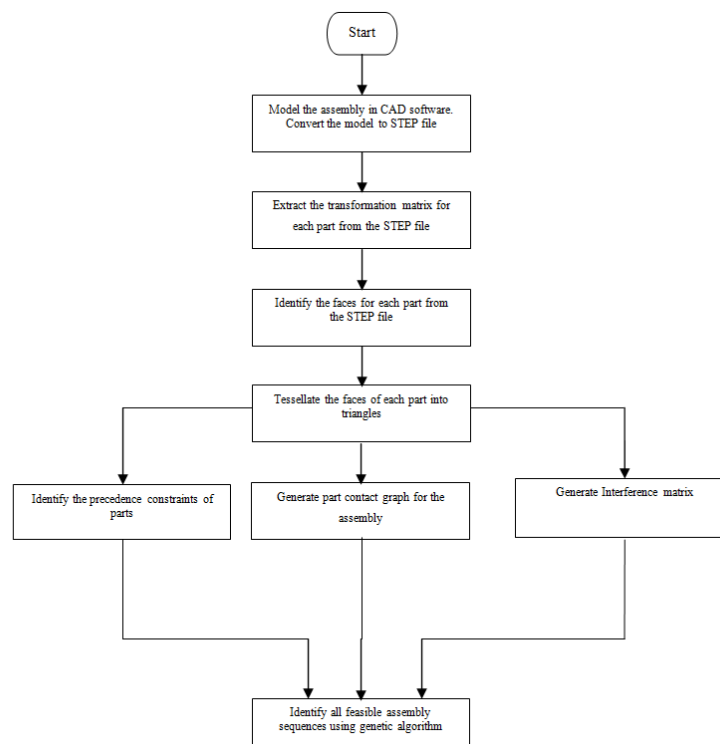


Figure 1 : Flowchart of the Method

In this work, the Fast, minimum storage Ray/triangle intersection algorithm developed by Tomas and Ben (1997) was utilized to find the obstruction of each part with all other parts. In the fourth step of the method, the part interference matrix is formulated. New set of rules are formulated to modify the interference matrix of part and to identify the precedence disassembly constraint for part. Finally, genetic algorithm is applied to find possible disassembly sequences.

PART CONTACT GRAPH

The Part contact graph formulated in this approach is a simple graph with undirected edges. It describes the relation between the parts of the assembly and can be defined as a 2-tuple: $G = \{P, E\}$. The node $P = \{p_1, p_2, \dots, p_m\}$ denotes the parts in the assembly. $E = \{e_1, e_2, \dots, e_k\}$ denotes the edges. An edge between the two vertices is formed if there is a contact between the parts. When the result of the ray firing between the parts has zero obstruction value, then the nodes are joined as a solid undirected edge. Screwed joint assembly model and part contact graph is shown in figure 3 & 4.

PART REMOVAL DIRECTION IDENTIFICATION

The Part interference matrix is a matrix in which the obstruction during the disassembly of a part with all other parts is represented. For $P = \{p_1, p_2, \dots, p_m\}$ parts in an assembly, m part interference matrices are required to be generated. The elements in the matrix are either “1” or “0”. The obstruction of one part with the other while removing it is represented as “1”. The absence of obstruction is represented as “0”. The six columns in the part interference matrix represent the part removal directions in $x+$, $x-$, $y+$, $y-$, $z+$ and $z-$ respectively. The rows in the matrix for the part p_j represent the parts in the assembly other than p_j . The ray tracing algorithm can find the intersection distance of one part with the other. From the results of ray firing, part interference matrix is formulated. To find a removal direction for a part in an assembly, rays have to be fired towards other parts in all six orthogonal directions to identify the removal directions. When the result of the rays

fired from a part has no hit point with other parts in the assembly, then the ray fired direction is considered as the removal direction of the part. It is easy to find the interference of every part in the assembly by using CAD data. However, the interference data alone is not sufficient to find disassembly sequences whenever the assembly has a part requiring two steps to remove or a visible part mated firmly with inside part.

IDENTIFYING DISASSEMBLY MOVE FOR VISIBLE PART REQUIRING TWO STEP DISASSEMBLY MOVE

A part which is visible or non- visible is identified using 2D assembly views. The hidden edges removed front, top and side views of the assembly are orthographically projected by setting every part with different line colour style. The inside part will not be found in the projected views and is considered as non- visible part.

The middle part in the screwed joint assembly is obstructed by top part in Z+ direction but they are placed at a greater distance apart (refer figure 3). Similarly, the machine screw shown in the figure 5(a) has interference with handle. Unlike the middle part in the screwed joint assembly, machine screw in machine screw assembly can be removed in two steps without hitting the handle as shown in figure 5(b).

The machine screw is constrained to move in one direction (Z+) due to concentric mating contact with the flange hole. The part is displaced to its length in Z+ direction and then the removal direction can be identified by the ray firing algorithm. Not all parts in assembly require two step movements to disassemble. If every part is checked for two step removal process, then more time will be consumed for computation. If-then rule -1 is formulated to save computational time by considering selected parts only.

If-then Rule-1

If a visible part is constrained to move in one direction with an obstructed at a distance greater than the size of the visible part by another party, then identify the second disassembly step after displacing it to a distance of its size in the allowed movement direction. If a part satisfies the rule 1, then the interference matrix will be updated by considering the first step direction as removal direction.

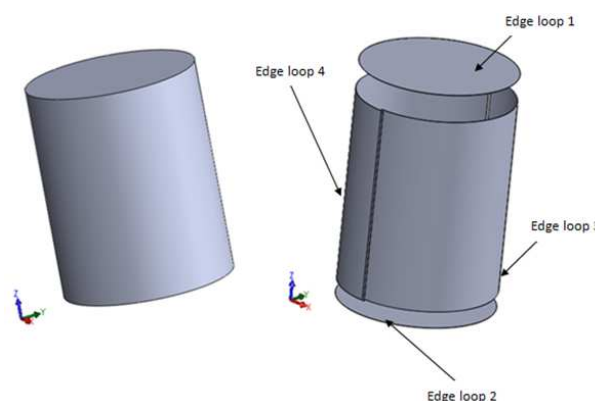


Figure 2 : Cylindrical Object

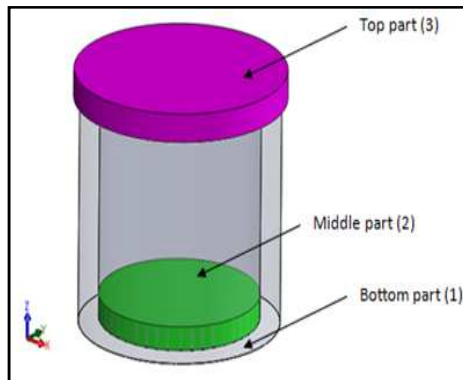


Figure 3

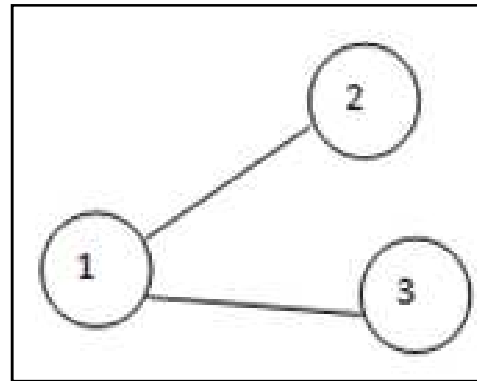


Figure 4 : Part Contract Graph

IDENTIFYING DISASSEMBLY MOVE FOR VISIBLE PART WHEN MATED WITH NON-VISIBLE PART

In some assembly, the non- visible part is firmly mated with visible part. The visible part cannot be removed until the non- visible part is approachable. For example, the double acting cylinder shown in figure 6 has an inside part (part 4). This part is connected with, part 3 by a screwed joint. The part 3 is designed to move forward and return in Z direction due to the pressurized fluid and It is moving through the hollow circular hole in parts 5,6 and 7. Although part 3 has no interference in Z- direction, it cannot be removed since it is connected firmly with the inside part (part number 4). Part 4 is accessible only when part 2 or parts 5, 6 and 7 are removed from the assembly. Removing either part 2 or parts 5, 6 and 7 from assembly will make part 4 visible. The precedent constraint parts for visible part are stored in a database. If –then rule 2 is formulated to identify the disassembly move conditions for the visible part while it is in mate with non- visible part.

If-then Rule-2

If a visible part is connected firmly with an inside part then the connected part is allowed to remove in no interference direction after the inside part is made visible.

IDENTIFYING REMOVAL CONDITION FOR CIRCLIP AND GIB HEADED COTTER

Li et al.,(2012) developed If-then rules for identifying disassembly precedence constrain for various connectors. Connectors like Circlip and Gib headed cotter are not investigated in their approach. If- then Rule is developed in this work to identify the removal directions for Circlip, Snap rings and Gib headed cotters. By analyzing the STEP AP 203 file of a part model, the cylinder or conical surface of extruded or cut feature is found out in this work.

If-then Rule-3

If a part is in concentric contact with another part and has interference matrix value of zero in all directions then move direction for the part is along the axis of cylindrical surface (see figure7).

If-then Rule-4

If a part has obstruction in all direction with a contact part, then move the part to the positive obstruction value and find the second move direction.

Part interference matrix will be modified based on rule 3. Strap type connecting rod end assembly is shown in figure 8. The interference matrix of part 5 of Strap type connecting rod end assembly is shown in table 1. Based on the rule

4, it is found that the part can be removed first in $x+$ direction and then followed by y ($y+$ or $y-$) direction only when parts 6 and 8 are not in the assembly. The precedent constraint parts are stored in a database.

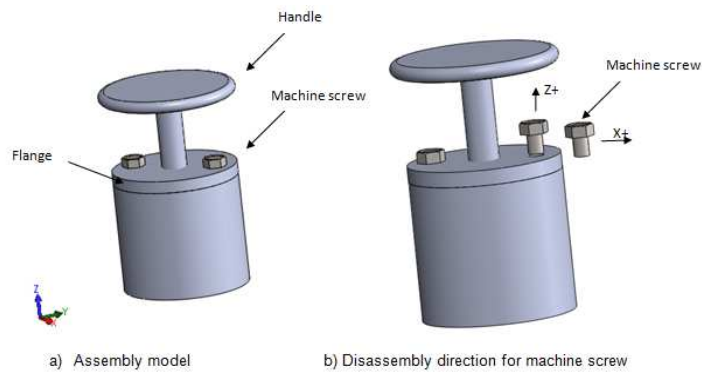


Figure 5 : Machine Screw Joint Assembly

Table 1 : Part Interference Matrix

part	Part Interference Direction					
	$x+$	$x-$	$y+$	$y-$	$z+$	$z-$
1	1	1	1	1	1	1
2	1	1	1	1	1	1
3	0	1	0	0	0	0
4	0	1	0	0	0	0
6	1	0	1	0	0	0
7	1	0	0	0	0	0
8	1	0	0	0	0	0

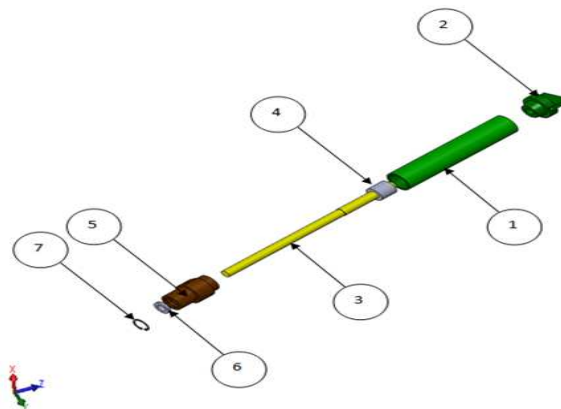


Figure 6: Exploded View of Double Acting Cylinder

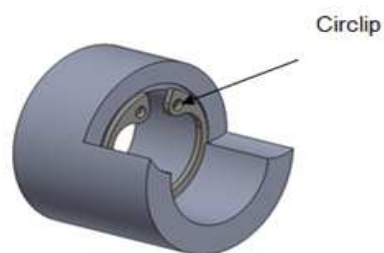


Figure 7: Circlip in an Assembly

GA-APPROACH TO FIND DISASSEMBLY SEQUENCES

GA is a highly robust search algorithm and it mimics the principle of natural genetics and natural selection to constitute search and optimization procedures. It is capable of exploring the search space widely and has been used successfully in solving NP-hard. The steps are followed in applying GA for any engineering applications are i). Encoding the chromosome ii) Develop the fitness function and iii)3. Reproduction, Crossover and Mutation.

In this method, the edges of the part contact graph are arranged as chromosome (\mathcal{X}). Each string in the chromosome represents the edge number of the graph. The edge numbers is arranged randomly and are not repeated. The strings in the chromosome represent the act of cutting the edges of the graph. If a node (part) has no edge contact with the graph then the part is considered to be separated from the assembly. Removal of all the edges of the graph will separate all the parts in the assembly into single individual parts. The removal of an edge can create two subassemblies or separating a part from the assembly or none of the part is removed.

The fitness function evaluates the quality of the chromosomes and ensures the feasible disassembly sequence generation. The fitness function is calculated in equation (1) and it is a summation of edge value edge cut penalty and precedence violation value.

$$\text{Fitness function} = (TC_1 + TC_2 + TC_3) \quad (1)$$

$$TC_1 = n * a1 \quad (2)$$

$$TC_2 = \sum_{x \in [n]} me(x) \text{ where } n = \{1, 2, 3, \dots, \text{no of edges}\} \quad (3)$$

$$x \rightarrow (p_i, p_j)$$

$$me(x) = \begin{cases} 0, \text{if } p_i \text{ and } p_j \text{ become isolated} \\ 0, \text{deg}(p_i) = 0, \text{deg}(p_j) > 0 \text{ and } ((\exists d) T(p_i, a)) = 0, \text{ where "a"} \\ \text{represents vertices in the path } p_j \text{ and } p_i \text{ becomes isolated vertex} \\ 0, \text{if } \text{deg}(p_i) > 0, \text{deg}(p_j) > 0, v1 \cap v2 = \emptyset \text{ and } ((\exists d) (\forall v1) T(v1, v2)) = 0; \\ \text{where } v1 \text{ is the vertices in the path of } p_i \text{ and } v2 \text{ is the vertices in the path of } p_j \\ 0, \text{if } \text{deg}(p_i) > 0 \text{ and } \text{deg}(p_j) > 0; p_i \text{ and } p_j \text{ are connected} \\ -k1, \text{otherwise} \end{cases} \quad (4)$$

$$k1 = (n - s1) * a2 \quad (5)$$

$$TC_3 = \begin{cases} -k1, \text{if removing part violates precedence constrain} \\ 0, \text{otherwise} \end{cases} \quad (6)$$

In this method the fitness function is formulated in such a way that the wrong choice of part removal or subassembly formation from the assembly/subassembly is penalized. If the disassembly sequence generation is carried out without any hindrance then the fitness value will be equal to TC_1 . The value TC_2 reduces value, if any, illegal value if any illegal move is followed. The reduction in fitness function is proportional to the position of the string. If the illegal move is carried earlier then the reduction by the TC_2 to the fitness function value is more. Whenever a part is removed before the precedence parts, then it is considered as an illegal move and the fitness function value for the chromosome is

deducted by a value equal to $k1$. The variation in the fitness value of the chromosome ensures the identification of best solutions.

All chromosomes with fitness value equal to the value of TC_1 will have no single illegal move. The edge value (TC_1) is calculated by using equation (2). The value is obtained by multiplying the number of edges in the part contact graph by edge value constant. The equation (3) is used to find TC_2 . For every cut of the edge of the graph, the value of $me(x)$ is calculated and the summation of each cut is assigned to TC_2 . The value of $me(x)$ is assigned based on five possibilities as given in equation (4). It is assigned zero value in the first condition when the assembly or subassembly with only two parts is divided. In the second condition, it is given a value of zero when a part is removed from an assembly or a subassembly without any obstruction. If an assembly or a subassembly is divided into subassemblies without any obstruction among them, then the value assigned to $me(x)$ is zero. The value assigned to $me(x)$ is also zero when an edge is deleted without removing a part or a subassembly. If an edge is cut in violation of any one of the four conditions, then the value assigned is $-k1$. The value of TC_2 is calculated by using equation (5).

Reproduction is a process in which individual chromosomes are selected to contribute for generating the offspring based on their fitness function values. The probability of selecting a string with a fitness value of f is f/ft , where ft is the sum of all of the fitness values in the population. Crossover is done to search for new solution by exploring new search areas in the solution space. The newly reproduced strings are mated at random. Each pair of strings undergoes crossing over at any integer position (selected randomly) along the string. In this method, single point crossover is followed. Mutation is an operation done to ensure that every region of the search space is reached. Mutation is performed randomly on any string of a chromosome. The $pcross$ and $pmut$ are crossover and mutation probability.

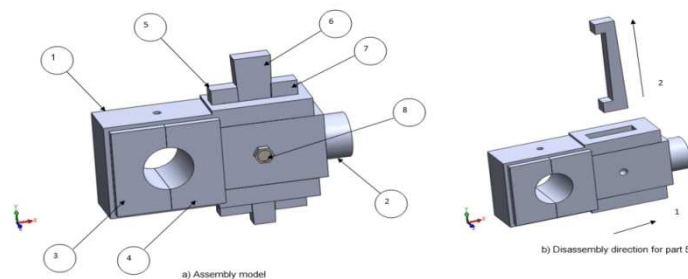


Figure 8: Strap Type Connecting Rd End Assembly

RESULTS AND DISCUSSIONS

The algorithm is implemented in MATLAB and the input data can be interactively altered in the program. The double acting which contains a circlip and also a visible part in threaded contact with non- visible part cylinder is taken as an example problem. The part contact graph of the double acting cylinder has 7 nodes and 12 edges. The assumed value of n , $a1$, $a2$, $pcross$ and $pmut$ are 12, 20, 2, 0.9, 0.01 respectively. By using rules 2 and 3, the disassembly sequences are identified. The table 2 presents the best removal sequence for individual parts in the assembly. From the table, it is found that for part numbers 2 and 7 can be removed first. Minimum disassembly operations needed for part 1, 6 and 4 is two only. Part numbers 5 and 6 requires three steps to disassemble. Complete disassembly can be achieved in six operations only. The disassembly sequence number 4 can divide the assembly into two subassemblies in first operation itself. Parts 1 and 2 constitute one assembly whereas the other subassembly consists of parts 3,4,5,6 and 7.

Table 2: Best Part Removal Sequences

S.No	Sequence	Operation						
		First	Second	Third	Four	Five	Six	Seven
1.	p2→p1→p4→p3→p7→p6→p5	Part 2	Part 1	Part 4	Part 3	Part 7	Part 6	Part 5
2.	P7→p6→p5→Sa4, 3 & Sa1,2→P4,3→P1,2	Part 7	Part 6	Part 5	Sa4,3 & Sa1,2	Part 4,3	Part 1,2	-
3.	P7→p6→p5→Sa4,3 & Sa1,2→P1,2→P4,3	Part 7	Part 6	Part 5	Sa1,2 & Sa4,3	Part 1,2	Part 4,3	-
4.	Sa1,2&Sa4,3,7,6,5→P4→P3→P7→P6→P5→P1,2	Sa1,2 & Sa4,3,7,6,5	Part 4	Part 3	Part 7	Part 6	Part 5	Part 1,2
5.	p2→Sa3,4&Sa1,5,6,7→p3,4→p1→p7→p6→p5	Part 2	Sa3,4 & Sa1,5,6,7	Part 4,3	Part 1	Part 7	Part 6	Part 5
6.	p2→Sa3,4&Sa1,5,6,7→p1→p7→p6→p5→p3,4	Part 2	Sa3,4 & Sa1,5,6,7	Part 1	Part 7	Part 6	Part 5	Part 4,3

CONCLUSIONS

In this method, disassembly sequences are identified by using STEP CAD files as input. Rules formed in this method helps in identifying the removal direction for parts requiring two step disassembly operations. This paper addresses the issue of identifying the subassemblies and generating disassembly sequences without human interaction. Extracting part surface information from STEP file facilitates the formation of a rule to identify the connectors like Snap rings and Circlip. The application of the approach to the illustrative example confirms the the superiority of the method in identifying all the possible disassembly sequences and the subassemblies.

REFERENCES

1. Bourjault, A. (1984). *Contribution to a methodological approach of automated assembly: automatic generation of assembly sequence*. University de Franche-Comte.
2. Cem, H., and Riaz, B. H., (2005), *An assembly sequence-planning system formechnical parts using neural network*, *Assembly Automation*, Vol.25, No.1, pp. 38-52.
3. DeFazio, TL., and Whitney, DE,(1987), "Simplified generation of all mechanical assemblysequences", *IEEE Journal on Robotics Automation*, Vol.3No.6, pp. 640–658.
4. Gottipolu, R., and Ghosh, K,(2003), *A simplified and efficient representation for evaluation and selection ofassembly sequences*, *Computers in Industry*, Vol.50, pp.251-264.
5. Homem De Mello. LS., and Sanderson AC.(1990), *AND/OR graph representation of assembly plans*, *IEEE Transactions on Robotics andAutomation*, Vol. 6, No.2, pp.188-199.
6. Homem de Mello, LS, and Sanderson, AC. (1991), *A correct and complete algorithm for the generation of mechanical assembly sequence*, *IEEE Transactions on Robotics andAutomation*, Vol 7, No.2, pp. 228–240.
7. Huang, W., & Xu, Q. (2017, August). *Automatic generation and optimization of stable assembly sequence based on ACO algorithm*. In *2017 IEEE International Conference on Mechatronics and Automation (ICMA)* (pp. 2057-2062). IEEE.
8. Kuo, TC.,(2000), *Disassembly sequence and cost analysis for electromechanical products*, *Robotics and Computer Integrated Manufacturing*, Vol.16, pp. 43–54.
9. Li, JR., Khoo, LP., and Tor, SB.(2002), *A hybrid disassembly sequence planning approach for maintenanc*, *Journal of Computing and Information Science in Engineering*, Vol. 2, pp.28–37.
10. LiJ., Wang, Q., Huang, P., and Shen, H.,(2012), *A novel connector - knowledge-based approach for disassembly precedence constraint generation* *International Journal of Advanced Manufacturing Technology*, Vol.9, pp. 293- 304.

11. Mok, SM., Ong K, and Wu CH,(2001), *Automatic Generation of Assembly Instructions using STEP*, *Proc. of IEEE International Conference on Robotics and Automation*, pp.313–318.
12. Murali, G. B., Sahu, P. K., Deepak, B. B. V. L., & Biswal, B. B. (2018, June). *Modified BAT Algorithm for Optimum Assembly Sequence Planning*. In *IOP Conference Series: Materials Science and Engineering* (Vol. 377, No. 1, p. 012091). IOP Publishing.
13. Ong, NS., and WongYC. (1999), *Automatic Subassembly Detection from a product Model for Disassembly Sequence Generation*, *International Journal of Advanced Manufacturing Technology*, Vol. 1, pp. 425-431.
14. Pan, C., Smith, SS., and Smith, GC.(2006), *Automated assembly sequence planning from STEP CAD files*, *International Journal of Production Research*, Vol. 19, No.8, pp.775-786.
15. Rabemanantsoa, M., and Pierce, S.(1996), *An artificial intelligence approach for generating assembly sequences in CAD/CAM*, *Artificial Intelligence in Engineering*, Vol.10, pp.97-107.
16. Raju, MVA, Bahubalendruni, B, Biswal, B & Deepak, BBVL (2016), *Optimal Robotic Assembly Sequence Generation Using Particle Swarm Optimization*, *Journal of Automation and Control Engineering*, Vol. 4, no. 2, pp.89-95.
17. Smith, GC., and Smith, SSF.(2002), *An enhanced genetic algorithm for automated assembly planning*, *Robotics and Computer Integrated Manufacturing*, Vol.18, No.5, pp.355–64.
18. Smith, S., Smith, G., and Chen, WH.(2012), *Disassembly sequence structure graphs: An optimal approach for multiple-target selective disassembly sequence planning*, *Advanced Engineering Information*, Vol. 26, pp.306-316.
19. Sung, RCW,(2001), *Automatic Assembly Feature Recognition and Disassembly Sequence Generation*, *Ph. D Thesis, Heriot-Watt University, Edinburgh, UK*.
20. Tomas, M., and Ben, T,(1997), *Fast, Minimum StorageRay/Triangle Intersection*, *Journal of Graphic Tools*, Vol.2, No.1, pp. 21-28.
21. Tseng, HE., and Li, RK.(1999), *A novel means of generating assembly sequences using the connector concept*,. *Journal of Intelligent Manufacturing*, Vol. 10, pp. 423–435.
22. Vigano, R., and Osorio-Gomez, G. (2012), *Assembly planning with automated retrieval of assembly Sequences from CAD model information*, *Assembly Automation*, Vol.32, No.4, pp. 3 47-360.
23. Xing, Y., Chen, G., Lai, X., Jin, S. and Zhou, J. (2007), *Assembly sequence planning of automobile bodycomponents based on liaison graph*, *Assembly Automation*, Vol. 27 No. 2, pp. 157-64.
24. Yin, Z., Ding, H., Li, H., and Xiong, Y.,(2003), *A connector based hierarchical approach to assembly sequence planning for mechanical assemblies*, *Computer Aided Design*, Vol. 35, pp.37–56.
25. Zha, X. F. (2004), *Planning for STEP-based electromechanicalassemblies: an integrated approach*, *International Journal of Computer Integrated Manufacturing*, Vol. 17, pp. 305-26.